



## VRTRIX Data Glove Cpp API Tutorial



Date	Modified by	Comments
2019-08-27	Guo	Init Version.
2019-09-05	Guo	Update for VRTRIX Data Gloves Pro

## 简介

## Introduction

VRTRIX™ 数据手套支持通过 C++ API 提供的动态链接库对手套数据进行访问，读取手套实时数据流，获取手部各关节姿态，硬件信息例如电池电量以及信号强等，同时可以控制手套掌心震动模块。

该动态链接库在 VS2017 下编译，同时支持 32 位（x86）和 64 位（x64）版本。手套硬件支持 VRTRIX Data Glove DK1, DK2, Pro。

## 系统要求

## System Requirements

- Windows 10 及以上版本。
- Visual Studio 2017。

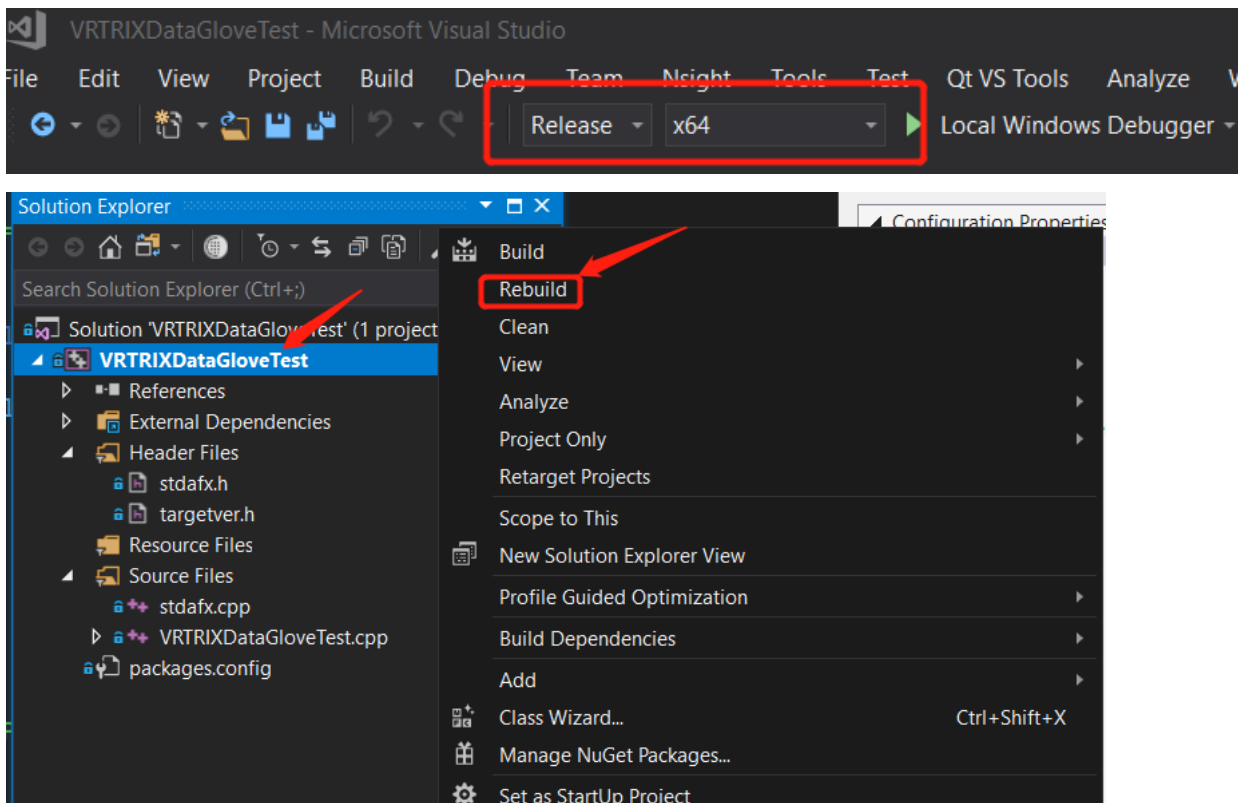
## 开发准备

## Prepare for Development

1. **数据手套配置：**请首先确保数据手套驱动软件已经成功安装，且数据手套已经成功配对，如果还未进行该操作，请按数据手套操作手册安装软件和操作后再进行下面的步骤。
2. **例程下载：**该动态链接库提供一个使用示例工程，可以从官方 [github](https://github.com/VRTRIX/VRTRIXGloveCppSDK) 网站下载工程源码：（<https://github.com/VRTRIX/VRTRIXGloveCppSDK>），该工程使用 VS2017 构建，默认 Windows SDK 版本为 10.0.17763.0。请使用 VS2017 打开，并修改 Windows SDK 版本为开发计算机上安装的 Windows SDK 版本，方能正常编译。

<b>General</b>	
Target Platform	Windows 10
Windows SDK Version	10.0.17763.0
Output Directory	\$(SolutionDir)\bin\x64\
Intermediate Directory	\$(Platform)\\$(Configuration)\
Target Name	\$(ProjectName)
Target Extension	.exe
Extensions to Delete on Clean	*.cdf;*.*cache;*.*obj;*.*obj.enc;*.*ilk;*.*ipdb;*.*iobj;*.*resources;*.*tlb;*.*tli;*.*tlh;*.*tmp;*.*rsp;*.*pg
Build Log File	\$(IntDir)\$(MSBuildProjectName).log
Platform Toolset	Visual Studio 2017 (v141)
Enable Managed Incremental Build	No
<b>Project Defaults</b>	
Configuration Type	Application (.exe)
Use of MFC	Use Standard Windows Libraries
Character Set	Use Unicode Character Set
Common Language Runtime Support	No Common Language Runtime Support
.NET Target Framework Version	
Whole Program Optimization	Use Link Time Code Generation
Windows Store App Support	No

3. 编译和运行例程：首先选择工程配置为 Release 模式，平台选择希望发布的平台，即 64 位计算机选择 x64， 32 位计算机选择 x86。右键选择工程文件，然后点击 Rebuild 进行编译。生成的可执行文件默认位置为\$(SolutionDir)\bin\\$(Platform)\，双击运行。



**4. 例程说明：**该例程演示了数据手套 C++ API 的初始化方法，包括回调类的注册与回调类的实现，同时演示了其中一些 API 函数的调用。完整的 API 函数接口请见 **VRTRIXGloveCppAPIReference.pdf**。以下是对例程中一些重要的函数和类的介绍：

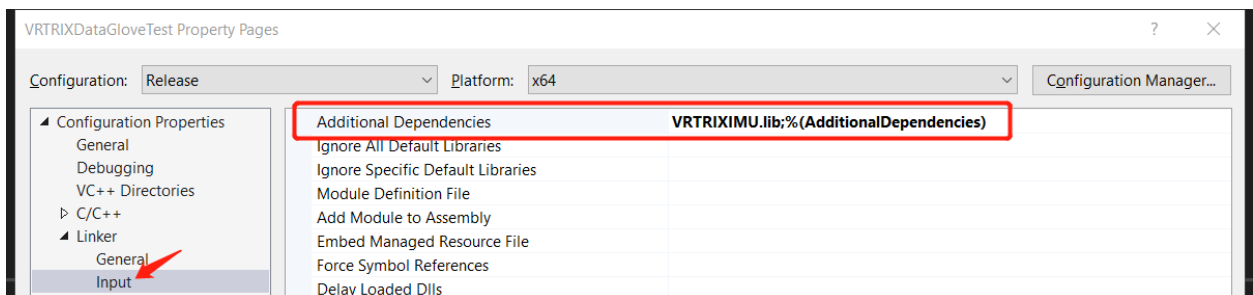
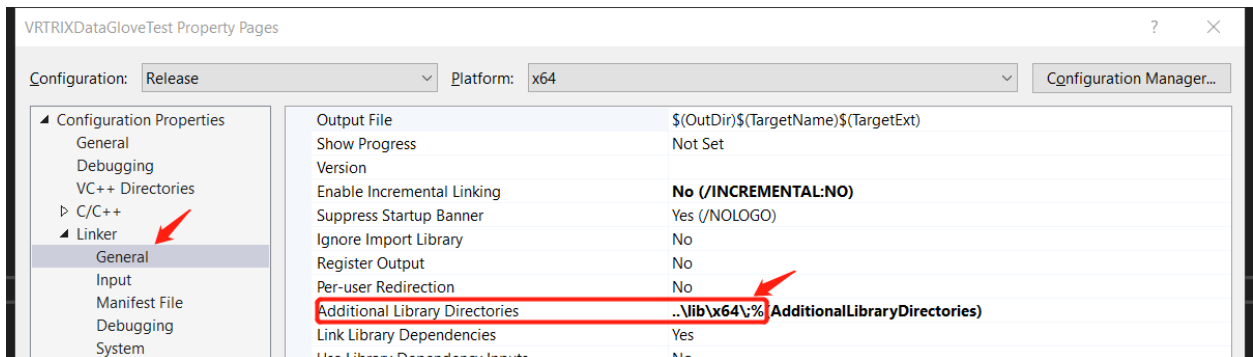
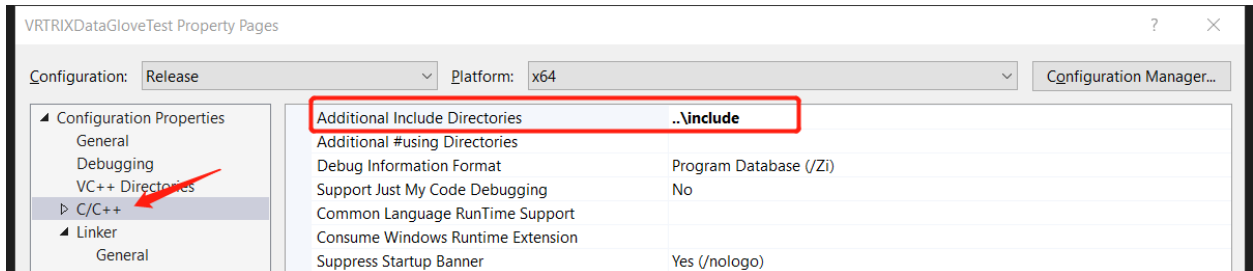
- **InitDataGloveStreaming:** 该函数包含了对数据手套所有的初始化流程，即先创建一个回调函数类的对象（**pEventHandler**），调用 API 中手套初始化函数（**InitDataGlove**）后，对该回调函数类对象的指针进行注册（**RegisterIMUDataCallback**）。随后打开端口（**OpenPort**），如果端口打开成功，则开始数据流传输（**StartDataStreaming**），否则关闭端口（**ClosePort**），且注销指针（**UnInit**）。
- **CVRTRIXIMUEventHandler:** 该类为回调函数接口类的实现，其继承于 **IVRTRIXIMUEventHandler** 为抽象类，回调函数以纯虚函数的形式存在，使用者必须要实现该抽象类中的所有函数，否则初始化时将无法注册回调函数。该类主要作用为接收动态链接库内部的信号和数据流。**OnReceivedNewPose** 回调函数在每接收到一个数据包时会被调用，故而所有对手套数据的处理都应写在这个函数的实现下。**OnReceivedNewEvent** 回调函数在每触发一个事件时会被调用，具体事件类型请看枚举类型 **HandStatus**，故而所有对手套事件的处理都应写在这个函数的实现下。

## 快速开始

### Quick Start

- 1. 新建工程：**首先打开 VS2017，新建一个控制台工程。然后在解决方案路径下新建 **include** 和 **lib** 文件夹。将 **IVRTRIXIMUEventHandler.h** 和 **VRTRIX\_IMU.h** 复制到 **include** 文件夹下，将对应平台下的 **VRTRIXIMU.lib** 复制到 **lib** 文件夹下。
- 2. 配置环境：**右键选择工程文件，选择 **Properties -> C/C++**，将 **Additional Include Directories** 改为 **..\include**，即将之前创建的 **include** 文件夹中的头文件进行包含。

右键选择工程文件，选择 **Properties** → **Linker**，将 **Additional Library Directories** 改为 `..\lib`，即将之前创建的 `lib` 文件夹加入搜索路径。然后选择 **Properties** → **Linker** → **Input**，将 `VRTRIXIMU.lib` 填入 **Additional Dependencies**。点击确认进行保存。



注意配置时请注意该属性页面上的工程配置和平台配置和需求的相符，例如上图为 **Release** 工程，目标平台 **64** 位。

### 3. 编写代码

模仿示例工程，如上所述，先对 `IVRTRIXIMUEventHandler` 类进行实现，然后依次调用 `InitDataGlove`，`RegisterIMUDataCallback`，`OpenPort` 和 `StartDataStreaming` 对数据手套进行初始化。随后对回调函数中获取到的数据或事件做进一步的处理。

### 4. 运行代码

代码编译完成后，需要将动态链接库 VRTRIXIMU.dll 复制到可执行文件路径下再运行，或者将动态链接库路径添加到系统环境变量 PATH 中，如果对 Windows 系统环境变量添加不熟悉，请参考如下文档：<https://www.computerhope.com/issues/ch000549.htm>

## 常见问题

### FAQ

#### 1. 为何编译提示 **error MSB8036**?

该报错提示为 Windows SDK 未找到，本示例 C++ 工程默认使用 Windows 10.0.17763.0 SDK，如果开发电脑上未安装，请选择其他已有的 SDK 版本，再重新编译。

#### 2. 为何编译提示 **fatal error C1083** 或者 **error LNK2001**?

C1083 为编译器报错，意为头文件未找到，请确认 Properties -> C/C++ -> Additional Include Directories 中包含了正确的头文件路径。

LNK2001 为链接失败，意为静态链接库未找到，请确认 Properties -> Linker -> Additional Library Directories 和 Properties -> Linker -> Input 下包含了正确的静态库。

首先确保两个 USB 接收器正常插在 PC 的 USB 接口上，其次检查 VRTRIXGloveDataStreaming 这个脚本的配置是否有不小心被篡改的地方。另外保证 VRTRIXGloveConfigTool 该客户端软件在 Unity Demo 运行时关闭。客户端配置软件只用于配置硬件参数，进行配对，以及运行 Demo，不是作为 Runtime 存在，开发时无需打开该软件。

#### 3. 为何编译成功后运行提示 **dll 未找到**?

代码编译完成后，需要将动态链接库 VRTRIXIMU.dll 复制到可执行文件路径下再运行，或者将动态链接库路径添加到系统环境变量 PATH 中，如果对 Windows 系统环境变量添加不熟悉，请参考如下文档：<https://www.computerhope.com/issues/ch000549.htm>

#### 4. 为何初始化手套完成后，数据流反复中断，且一直显示 **Righthand/Lefhand setup to channel x?**

请确保在进行 C++ 程序开发之前，先安装并运行过客户端配置软件 VRTRIXGloveConfigTool，如果未安装请参考文档“北京无远弗届数据手套操作手册”先安装客户端软件对硬件驱动进行正确的安装和初始化配置。

如果已经安装和运行过客户端软件，依然有卡顿或者断开的情况，将手套 USB 接收器通过 USB Hub 接出，避免 PC 相邻 USB 口上有大流量的硬件设备例如无线网卡，移动硬盘等，同时避免手套 USB 接收器和手套之间没有过多的遮挡。在正常电磁环境下，手套的正常通信距离应在 10m 以上。

#### 5. 为何运行示例程序提示 **RIGHT/LEFT HAND PORT OPEN FAILED!**

此报错原因是硬件端口占用，请确定在运行示例程序前，已经关闭 VRTRIXGloveConfigTool 客户端软件，且已经断开其他平台 SDK 对硬件的连接，例如 UNITY, UE4 或者 MotionBuilder。关闭后再运行示例程序即可。

#### 6. 初始化函数 **InitDataGlove** 中 **mode** 和 **version** 这两个形参是什么含义？

首先 version 代表硬件版本，目前 C++ SDK 支持 DK1，DK2 和 PRO，默认采用 DK2，如果不确定硬件版本，请联系销售代表。

另外 mode 形参为初始化模式，默认模式为 InitMode\_Normal 普通模式。如果想解锁五指张开，即航向角测定，可以使用 InitMode\_Advanced，也可以先用 InitMode\_Normal 模式初始化，在运行时调用 SwitchToAdvancedMode 函数切换为 InitMode\_Advanced。InitMode\_VR 同时会对 VR 系统进行初始化，如果没有 VR 系统（默认采用 HTC 虚拟现实头显和 SteamVR 平台），请不要选择这个模式。

#### 7. 为何在 **InitMode\_Advanced** 模式下测定手指航向夹角时，有时输出会出现跳变？

请远离磁性物体，包括电脑系统，铁架桌椅，耳机，手机等磁场辐射源，远离 1m 以上然后戴着手套划八字，八字校准 1 分钟。打开客户端软件运行 3D demo 看手部形态稳定下来后，点击 UI 左侧的 Hardware Calibrate 对当前磁环境参数进行储存。该操作只需要进行一次，以后在相同环境下无需再进行校准。